Garage Vol 2

WORKING WITH TEST PRACTICE

SEZGIN, NURI

Table of Contents

Purpose

The first chapter of a Garage Application view, presenter, service and repository have already implemented. Now, modifications will be applied to the data layer. The file provider should be removed and replaced current data provider with MySQL DB.

Sample

Tips

>>>> * >>>> indicates to program lifecycle.
>> refers to program's outputs.
<< shows user's inputs.

I/O

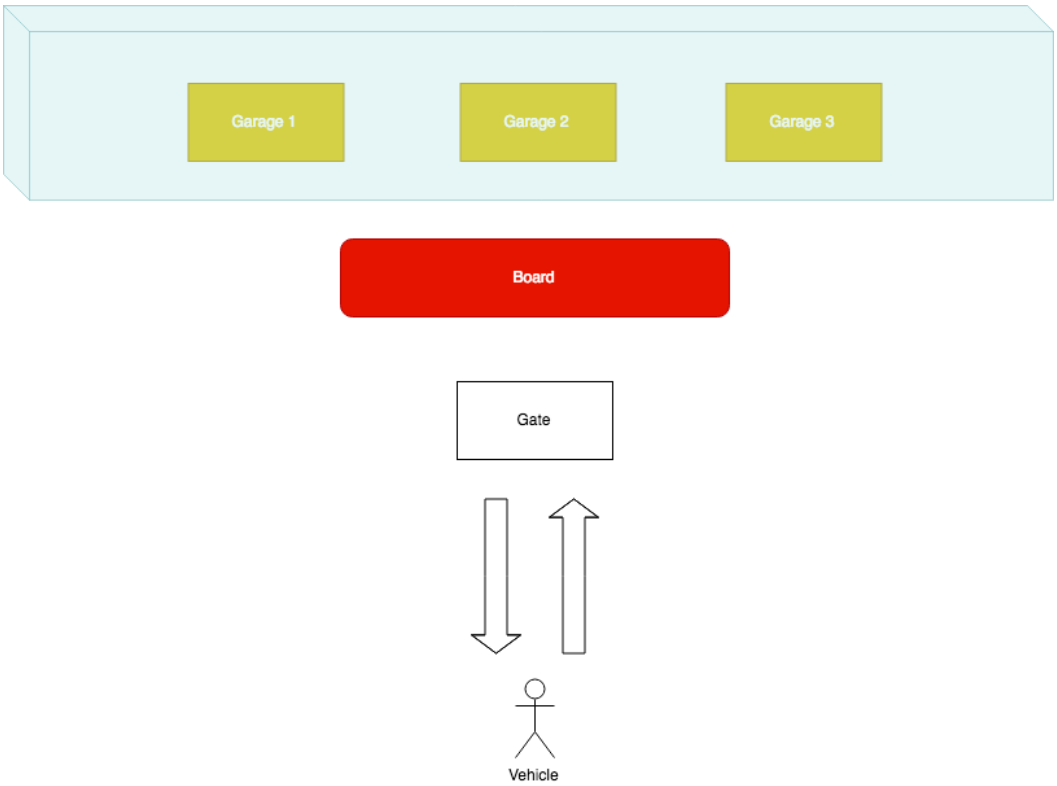
>>>> Program Start >>>>
>> Menu
1) Press "1" for add a vehicle.
2) Press "2" for removed a vehicle.
3) Press "3" for print board content.
4) Press "4" for quit the application.
5) Press "?" for print that menu.

>> Choose option:
<< 1
>> Pressed option is "1".
>> License plate:
<< 34TU123
>> Brand:
<< BMW
>> Model:
<< M3
>> Year:
<< 2013
>> Color:
<< Blue
>> Location of BMW M3 34TU123 is Garage1.
>> Board Garage1: 9, Garage2: 10, Garage3: 10
>> Choose option:
<< 2
>> Pressed option is "2".
>> Enter a license plate:
<< 34TU123
>> BMW M3 34TU123 is removed from Garage1.
>> Board Garage1: 10, Garage2: 10, Garage3: 10
>> Choose option:
<< 4
>>>> Program End >>>>

Program Design

Introduction

Following diagram shows the program's components. A vehicle comes from outside; the board shows the availability of garages and a vehicle should be navigated to an appropriate area. For different case, a vehicle goes out from a garage; the board should be refreshed.



Garage

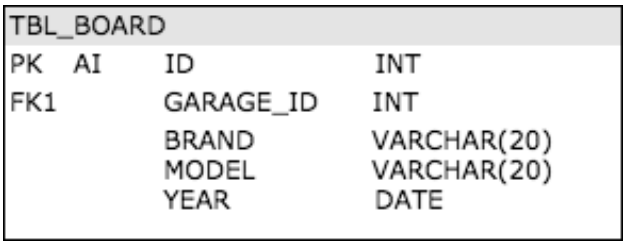
The program must have three different garage instances. Each garage information should be stored in "TBL_GARAGE" table and also garage values should be specified as a name, capacity, and availability for each occurrence.

Following ER diagram shows content of the "TBL_GARAGE".

TBL_GARAGE		
PK AI	ID	INT
	NAME	VARCHAR(80)
	CAPACITY .	INT(3)
	AVAILABILITY	INT(3)

Board

The “TBL_BOARD” should be kept all parked vehicle instances. Below diagram is showing “TBL_BOARD” fields and names.



Vehicle

A parked vehicle should be initialized with unique values license plate, brand, model, year, color and located in where.

Relationship Between Tables

Acknowledgments

Name	**Description**
PK	*Primary Key*
FK	*Foreign Key*
AI	*Auto Increment*
VARCHAR	*Collection of Characters*
DATE	*Database Date (yyyy-MM-dd)*

Functional Requirements

1. Program
    a. The program must be run infinitely.
    b. On the first run, the application menu should be printed.
    c. A menu content should be as same as below.
    Menu
        1) *Press "1" for add a vehicle.*
        2) *Press "2" for removed a vehicle.*
        3) *Press "3" for print board content.*
        4) *Press "4" for quit the application.*
        5) *Press "?" for print that menu.*
    d. User input should be asked as like below.
    Choose option:
    e. The selected option should be printed. Like that after this.
    Pressed option is "1".
    f. For option "1",
        1) License plate, brand, model, color should be requested to a user. A sample for collection of inputs.
        2) After adding the new vehicle to garage, the output should be printed.
        3) If the vehicle' s license plate is not valid or already any of garage has a vehicle with that license plate, "Error: license plate is not correct, you must enter valid license plate" message should be shown.
        4) Now a user can add or remove vehicle, print board status or menu, terminate the program.
    g. For option "2",
        1) License plate should be given by user. The sample is above.
        2) A removed vehicle should be shown on console.
        3) If the vehicle' s license plate is not valid or already any of garage has not a vehicle with that license plate, "Error: license plate is not correct, you must enter valid license plate" message should be shown.
        4) Now a user can add or remove vehicle, print board status or menu, terminate the program.
    h. For option "3",
        1) The board status should be printed to console out.
        2) Now a user can add or remove vehicle, print board status or menu, terminate the program.
    i. For option "4",
        1) The program must be terminated.
        2) Now a user **can't** add or remove vehicle, print board status or menu, terminate the program.
    j. For option "?",
        1) The menu should be shown.
        2) Now a user can add or remove vehicle, print board status or menu, terminate the program.

MySQL Database

MySQL DB should be installed on Docker. A practice about that is currently available on InnerSource.